IN THE CLAIMS

- 1. (Currently Amended) A multiprocessor system,
 comprising:
- a processing sub-system including a plurality of processors and a processor memory system;

a scalable network operable to couple the processing subsystem to an input/output (I/O) sub-system;

the I/O sub-system including a plurality of I/O interfaces;

the I/O interfaces each operable to couple a peripheral device to the multiprocessor system and to store copies of data from the processor memory system in the local cache for use by the peripheral device; and

. a coherence domain comprising the processors and processor memory system of the processing sub-system and the local caches of the I/O sub-system, the local caches of the I/O subsystem operable to participate in the coherence domain through timed access to data in the processor memory system;

wherein the I/O subsystem is operable to invalidate data upon expiration of a time interval and the processor memory system is operable to identify the data as free upon expiration of the time interval without sending any invalidation messages.

- 2. (Original) The multiprocessor system of Claim 1, wherein at least one of the I/O interfaces comprises a Peripheral Component Interconnect (PCI) interface.
- 3. (Original) The multiprocessor system of Claim 1, wherein the scalable network comprises a plurality of routers.

- 4. (Original) The multiprocessor system of Claim 1, the processor memory system comprising a plurality of discrete processor memories.
- 5. (Original) The multiprocessor system of Claim 4, wherein the discrete processor memories are each dedicated to a processor.
- 6. (Original) The multiprocessor system of Claim 1, the processor memory system including a directory operable to identify data cached in an I/O interface.
- 7. (Original) The multiprocessor system of Claim 6, the processor memory system operable to invalidate a copy of data stored in a local cache of an I/O interface in response to a request for the data by a processor.
- 8. (Original) The multiprocessor system of Claim 1, the I/O interfaces each operable to pre-fetch data from the processor memory and to store the data in the local cache for use by a corresponding peripheral device.

9. (Currently Amended) A method for maintaining data at input/output (I/O) interfaces of a multiprocessor system, comprising:

coupling a plurality of processors to a processor memory system;

coupling a plurality of I/O interfaces to the processor memory;

coupling a peripheral device to each I/O interface;

caching copies of data in the processor memory system in the I/O interfaces for use by the peripheral devices; and

maintaining coherence between the copies of data in the I/O interfaces and data in the processor memory system, the I/O interfaces operable to participate in coherence through timed access to the data in the processor memory system;

invalidating the data at the I/O interfaces upon expiration of a time interval;

identifying the data as free in the processor memory system upon expiration of the time interval, invalidating and identifying the data being performed without sending any invalidation messages.

- 10. (Original) The method of Claim 9, wherein at least one of the I/O interfaces comprises a Peripheral Component Interconnect(PCI) interface.
- 11. (Original) The method of Claim 9, further comprising coupling the I/O interfaces to the processor memory system through a scalable network.

- 12. (Original) The method of Claim 9, further comprising coupling the I/O interfaces to the processor memory system through a scalable network comprising a plurality of routers.
- 13. (Original) The method of Claim 9, wherein the processor memory system comprises a plurality of discrete processor memories.
- 14. (Original) The method of Claim 13, wherein each discrete processor memory is dedicated to a processor.
- 15. (Original) The method of Claim 9, further comprising identifying in the processor memory system data having a copy cached in the I/O interfaces.
- 16. (Original) The method of Claim 15, further comprising:

invalidating a copy of data cached in an I/O interface in response to a request by a processor for the data; and

releasing the data to the processor after invalidation of the copy in the \dot{I}/\dot{O} interface.

17. (Original) The method of Claim 9, further comprising:

pre-fetching data from the processor memory system; and caching the data in an I/O interface for use by a corresponding peripheral device.

- 18. (Currently Amended) An input/output (I/O) interface, comprising:
 - a peripheral communications port;
- a local cache operable to store copies of data from a processor memory coupled to the I/O interface; and
- a resource manager operable to invalidate outdated data from the local cache to maintain coherence with the processor memory without receiving an invalidation message.
- 19. (Original) The I/O interface of Claim 18, wherein the interface is a Peripheral Component Interconnect (PCI) interface.
- 20. (Original) The I/O interface of Claim 18, the resource manager further operable to invalidate copies of data from the local cache in response to instructions from the processor memory.
- 21. (Original) The I/O interface of Claim 18, wherein a copy of data is outdated upon expiration of a time period for storage of the copy.

22. (Currently Amended) A method for interfacing a peripheral device with a multiprocessor system, comprising:

storing copies of data in a processor memory system of the multiprocessor system in a local I/O memory at an interface for the peripheral device; and

maintaining coherence between data in the processor memory system and copies of data in the local I/O memory, the local I/O memory operable to participate in coherence through timed access to the data in the processor memory system;

invalidating the data at the I/O interfaces upon
expiration of a time interval;

identifying the data as free in the processor memory system upon expiration of the time interval, invalidating and identifying the data being performed without sending any invalidation messages.

- 23. (Original) The method of Claim 22, further comprising maintaining coherence between data in the processor memory system and copies of data in the local I/O memory by at least removing outdated copies of data from the local I/O memory.
- 24. (Original) The method of Claim 22, further comprising maintaining coherence between data in the processor memory system and copies of data in the local I/O memory by at least removing copies of data from the local I/O memory in response to instructions from the processor memory system.
- 25. (Original) The method of Claim 23, wherein a copy of data is outdated upon expiration of a time period for storage of the copy.